

Notice of Allowability

Application No.

10/689,417

Examiner

INSUN KANG

Applicant(s)

AGARWAL ET AL

Art Unit

2193

- The MAILING DATE of this communication appears on the cover sheet with the correspondence address--

All claims being allowable, PROSECUTION ON THE MERITS IS (OR REMAINS) CLOSED in this application. If not included herewith (or previously mailed), a Notice of Allowance (PTOL-85) or other appropriate communication will be mailed in due course. **THIS NOTICE OF ALLOWABILITY IS NOT A GRANT OF PATENT RIGHTS.** This application is subject to withdrawal from issue at the initiative of the Office or upon petition by the applicant. See 37 CFR 1.313 and MPEP 1308.

1. ☒ This communication is responsive to 1/10/2008.
2. ☒ The allowed claim(s) is/are 1-4, 6-10, and 12-35 (renumbered as 1-33).
3. ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 - a) ☐ All b) ☐ Some* c) ☐ None of the:
 1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this national stage application from the International Bureau (PCT Rule 17.2(a)).

* Certified copies not received: _____.

Applicant has THREE MONTHS FROM THE "MAILING DATE" of this communication to file a reply complying with the requirements noted below. Failure to timely comply will result in ABANDONMENT of this application.

THIS THREE-MONTH PERIOD IS NOT EXTENDABLE.

4. ☐ A SUBSTITUTE OATH OR DECLARATION must be submitted. Note the attached EXAMINER'S AMENDMENT or NOTICE OF INFORMAL PATENT APPLICATION (PTO-152) which gives reason(s) why the oath or declaration is deficient.
 5. ☐ CORRECTED DRAWINGS (as "replacement sheets") must be submitted.
 - (a) ☐ including changes required by the Notice of Draftsperson's Patent Drawing Review (PTO-948) attached
 - 1) ☐ hereto or 2) ☐ to Paper No./Mail Date _____.
 - (b) ☐ including changes required by the attached Examiner's Amendment / Comment or in the Office action of Paper No./Mail Date _____.
- Identifying indicia such as the application number (see 37 CFR 1.84(c)) should be written on the drawings in the front (not the back) of each sheet. Replacement sheet(s) should be labeled as such in the header according to 37 CFR 1.121(d).
6. ☐ DEPOSIT OF and/or INFORMATION about the deposit of BIOLOGICAL MATERIAL must be submitted. Note the attached Examiner's comment regarding REQUIREMENT FOR THE DEPOSIT OF BIOLOGICAL MATERIAL.

Attachment(s)

1. ☒ Notice of References Cited (PTO-892)
2. ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3. ☐ Information Disclosure Statements (PTO/SB/08),
Paper No./Mail Date _____
4. ☐ Examiner's Comment Regarding Requirement for Deposit of Biological Material
5. ☐ Notice of Informal Patent Application
6. ☒ Interview Summary (PTO-413),
Paper No./Mail Date 20080226.
7. ☒ Examiner's Amendment/Comment
8. ☒ Examiner's Statement of Reasons for Allowance
9. ☐ Other _____.

/Lewis A. Bullock, Jr./

EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR

1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

Authorization for this examiner's amendment was given in a telephone interview with Mr. Frederick Gibb (reg. 37,629) on 2/26 and 29, 2008.

2. The application has been amended as follows:

These amendments were necessary in order to further clarify the claims and obviate any rejection under 35 U.S.C. 112 2nd and 101.

1. (Currently Amended) A method, implemented in a data processing system, for determining run-time dependencies between logical components of a data processing environment, the method comprising the steps of:

monitoring run-time activity of each of a first logical component and a second logical component of the data processing environment, said monitoring comprising determining a first activity period for said first logical component and a second activity period for said second logical component, wherein said determining comprises determining a first start time and a first end time for said first logical component and a second start time and a second end time for said second logical component;

comparing the monitored run-time activity of the first logical component with the monitored run-time activity of the second logical component to identify correlations between the

monitored run-time activity of the first and second logical components, said comparing comprising comparing said first activity period and said second activity period to identify correlations between said first logical component and said second logical component, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

in response to identification of a positive correlation between the monitored run- time activity of the first and second logical components, recording the existence of a dependency relationship between the first and second logical components, wherein the step of monitoring the run-time activity of said first logical component and said second logical component is performed by a monitoring agent and comprises generating events by the monitoring agent in response to completion of the processing of poll requests by each of the first logical component and said second logical component[[s]], and wherein the step of comparing the monitored run- time activity comprises:

calculating an activity period for each of said first logical component and said second logical component[[s]] in response to the generated events indicating the completion of processing of [[a]] the poll requests by the respective one of said first logical component[[s]] and said second logical component[[s]]; and
determining whether the activity period of said first logical component contains the activity period of said second logical component.

8. (Currently Amended) The method according to claim 1, all the limitations of which are incorporated herein by reference, further comprising:

monitoring run-time activity for a plurality of executions of each of [[a]] the first and
[[a]] the second logical component, comparing the monitored run-time activity for the plurality
of executions of the second logical component with the monitored run-time activity for the
plurality of executions of the first logical component;

determining a proportion of executions of the first logical component for which a positive
correlation is identified between the compared run-time activity of the first and second logical
components; and

recording in association with the recorded dependency relationship a value representing
the determined proportion of executions of the first logical component for which a positive
correlation is identified.

9. (Currently Amended) The method according to claim 1, all the limitations of
which are incorporated herein by reference, further comprising:

monitoring run-time activity for a plurality of executions of each of [[a]] the first and
[[a]] the second logical component,

comparing the monitored run-time activity for the plurality of executions of the second
logical component with the monitored run-time activity for the plurality of executions of the first
logical component;

determining a proportion of executions of the second logical component for which a
positive correlation is identified between the compared run-time activity of the first and second
logical components; and

recording in association with the recorded dependency relationship a value representing
the determined proportion of executions of the second logical component for which a positive

correlation is identified.

14. (Currently Amended) The method according to claim 1, all the limitations of which are incorporated herein by reference, wherein the events are generated in response to [[a]] the monitoring agent polling the first and second components for information relating to the processing of requests.

16. (Currently Amended) The method according to claim 1, all the limitations of which are incorporated herein by reference, for monitoring a data processing system which comprises a monitoring interface for accessing run-time activity data within the data processing system, wherein the step of monitoring run-time activity comprises [[a]] the monitoring agent accessing run-time activity data via the monitoring interface.

20. (Currently Amended) A method of fault management comprising the steps of:
monitoring run-time activity of each of a first logical component and a plurality of additional logical components of a data processing environment, said monitoring comprising determining a first activity period for said first logical component and additional activity periods for said additional logical components, wherein said determining comprises determining a first start time and a first end time for said first logical component and additional start times and additional end times for said additional logical components;

comparing the monitored run-time activity of the first logical component with the monitored run-time activity of each of the plurality of additional logical components, to identify positive correlations between the monitored run-time activity of the first logical

component and the monitored run-time activity of any of the plurality of additional logical components, said comparing comprising comparing said first activity period and said additional activity periods to identify correlations between said first logical component and any of said additional logical components, wherein said comparing comprises determining whether said first start time is before said additional start times and whether said first end time is after said additional end times;

recording the existence of a dependency relationship between the first logical component and any of the plurality of additional logical components for which a positive correlation is identified;

aggregating the recorded dependency relationships to determine a set of logical components having dependency relationships with the first logical component; and

responding to identification of a problem affecting the first logical component by analyzing the [[set]] additional [[of]] logical components having dependency relationships with the first logical component,

wherein the step of monitoring the run-time activity of said first logical component and said additional logical ~~component~~ components is performed by a monitoring agent and comprises generating events by the monitoring agent in response to completion of the processing of poll requests by each of said first logical component and said additional logical components, and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said additional logical components in response to the generated events indicating the completion of processing of [[a]] the poll requests by the respective one of said first logical component and said additional logical components; and

determining whether the activity period of said first logical component contains the activity period of said additional logical ~~component~~ components.

21. (Currently Amended) The method according to claim 20, all the limitations of which are incorporated herein by reference, wherein the step of aggregating the recorded dependency relationships comprises sorting the dependencies into an order determined by a sorting heuristic, and the step of analyzing the [[set]] additional [[of]] logical components comprises analyzing components of the [[set]] additional [[of]] logical components in said determined order.

22. (Currently Amended) A method for determining a set of logical components of a data processing environment which are likely to be affected by termination of a first logical component of the data processing environment, the method comprising:

monitoring run-time activity of each of a first logical component and a plurality of additional logical components of a data processing environment, said monitoring comprising determining a first activity period for said first logical component and additional activity periods for said additional logical components, wherein said determining comprises determining a first start time and a first end time for said first logical

component and additional start times and additional end times for said additional logical components;

comparing the monitored run-time activity of the first logical component with the monitored run-time activity of each of the plurality of additional logical components, to identify positive correlations between the monitored run-time activity of the first logical component and the monitored run-time activity of any of the plurality of additional logical components, said comparing comprising comparing said first activity period and said additional activity periods to identify correlations between said first logical component and any of said additional logical components, wherein said comparing comprises determining whether said first start time is before said additional start times and whether said first end time is after said additional end times; and

recording the existence of a dependency relationship between the first logical component and any of the plurality of additional logical components for which a positive correlation is identified; and

aggregating the recorded dependency relationships to determine [[set]] additional [[of]] logical components having dependency relationships with the first logical component,

wherein the step of monitoring the run-time activity of said first logical component and said additional logical ~~component~~ components is performed by a monitoring agent and comprises generating events by the monitoring agent in response to completion of the processing of poll requests by each of said first logical

component and said additional logical components, and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said additional logical components in response to the generated events indicating the completion of processing of [[a]] the poll requests by the respective one of said first logical component and said additional logical components; and

determining whether the activity period of said first logical component contains the activity period of said additional logical ~~component~~ components.

23. (Currently Amended) A computer program product comprising a recordable storage medium having program code ~~recorded on a recording medium~~, for controlling the operation of a data processing system on which the program code executes to determine run-time dependencies between logical components of a data processing environment, the program code comprising:

at least one monitoring agent for monitoring run-time activity data of a first logical component and a second logical component~~[[s]]~~ of the data processing environment, said monitoring comprising determining activity periods for said first and second logical components, and for sending the monitored run-time activity data to a correlation identifier;

a correlation identifier for receiving, from the at least one monitoring agent, monitored run-time activity data of each of [[a]] the first logical component and [[a]] the second logical component of a data processing environment, wherein said receiving comprises receiving a first start time and a first end time for said first logical component and a second start time and a second end time for said second logical component, and for comparing the monitored run-time

activity data of the first logical component with the monitored run-time activity data of the second logical component to identify positive correlations between the monitored run-time activity of the first and second logical components, said comparing comprising comparing said activity periods of said first and second logical components to identify correlations between said first logical component and said second logical component, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

a dependency generator for responding to identification of a positive correlation between the monitored run-time activity of the first and second logical components by recording the existence of a dependency relationship between the first and second logical components,

wherein said monitoring run-time activity data of the first and second logical components comprises generating events by the at least one monitoring agent in response to completion of the processing of poll requests by each of the first and second logical components, and wherein said comparing the monitored run-time activity comprises:

calculating an activity period for each of the first and second logical components in response to the generated events indicating the completion of processing of [[a]] the poll requests by the respective one of the first and second logical components; and

determining whether the activity period of [[a]] the first logical component contains the activity period of [[a]] the second logical component.

24. (Currently Amended) A computer program product comprising a recordable storage medium having program code recorded on a recording medium, for controlling the operation of a data processing system on which the program code executes to determine run-time dependencies between logical components of a data processing environment, the program code comprising:

a correlation identifier:

for receiving, from at least one monitoring agent, monitored run-time activity data of each of a first logical component and a second logical component of a data processing environment, said receiving comprising receiving a first activity period for said first logical component and a second activity period for said second logical component, wherein said receiving comprises receiving a first start time and a first end time for said first logical component and a second start time and a second end time for said second logical component, and

for comparing the monitored run-time activity data of the first logical component with the monitored run-time activity data of the second logical component to identify positive correlations between the monitored run-time activity of the first and second logical components, said comparing comprising comparing said first activity period and said second activity period to identify correlations between said first logical component and said second logical component, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

a dependency generator for responding to identification of a positive correlation between the monitored run-time activity of the first and second logical components by generating a representation of the existence of a dependency relationship between the first and second logical components,

wherein ~~said at least one monitoring agent~~ monitoring monitors the run-time activity of said first logical component and said second logical component ~~performed by said monitoring agent comprises generating and generates~~ events in response to completion of the processing of poll requests by each of said first logical component and said second logical component[[s]], and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said second logical component[[s]] in response to the generated events indicating the completion of processing of ~~[[a]]~~ the poll requests by the respective one of said first logical component and said second logical component[[s]]; and

determining whether the activity period of said first logical component contains the activity period of the second component.

25. (Currently Amended) A data processing apparatus comprising:

a data processing unit;
a data storage unit;
a correlation identifier

for receiving, from at least one monitoring agent, monitored run-time activity data of each of a first logical component and a second logical component of a data

processing environment, said receiving comprising receiving a first activity period for said first logical component and a second activity period for said second logical component, wherein said receiving comprises receiving a first start time and a first end

time for said first logical component and a second start time and a second end time for said second logical component, and

for comparing the monitored run-time activity data of the first logical component with the monitored run-time activity data of the second logical component to identify positive correlations between the monitored run-time activity of the first and second logical components, said comparing comprising comparing said first activity period and said second activity period to identify correlations between said first logical component and said second logical component, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

a dependency generator for responding to identification of a positive correlation between the monitored run-time activity of the first and second logical components by recording in the data storage unit the existence of a dependency relationship between the first and second logical components,

wherein monitoring the run-time activity of said first logical component and said second logical component performed by said at least one monitoring agent comprises generating events by the at least one monitoring agent in response to completion of the processing of poll

requests by each of said first logical component and said second logical component[[s]], and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said second logical component[[s]] in response to the generated events indicating the completion of processing of [[a]] the poll requests by the respective one of said first logical component and said second logical component[[s]]; and

determining whether the activity period of said first logical component contains the activity period of the second component.

27. (Currently Amended) A distributed data processing system comprising:

a first data processing apparatus comprising a set of logical components to be monitored, and at least one monitoring agent for monitoring run-time activity data for the set of logical components and for sending the monitored run-time activity data to a correlation identifier on a second data processing apparatus, said monitoring comprising determining activity periods for said logical components; and

a second data processing apparatus comprising: a data processing unit; a data storage unit; a correlation identifier

for receiving, from the at least one monitoring agent, monitored run-time activity data of each of a first logical component and a second logical component of the set of logical components of a data processing environment, said receiving comprising receiving a first activity period for said first logical component and a second activity period for said second logical

component, wherein said receiving comprises receiving a first start time and a first end time for said first logical component and a second start time and a second end time for said second logical component, and

for comparing the monitored run-time activity data of the first logical component with the monitored run-time activity data of the second logical component to identify positive correlations between the monitored run-time activity of the first and second logical components, said comparing comprising comparing said first activity period and said second activity period to identify correlations between said first logical component and said second logical component, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

a dependency generator for responding to identification of a positive correlation between the monitored run-time activity of the first and second logical components by recording in the data storage unit the existence of a dependency relationship between the first and second logical components,

wherein monitoring the run-time activity of said first logical component and said second logical component performed by said at least one monitoring agent comprises generating events by the at least one monitoring agent in response to completion of the processing of poll requests by each of said first logical component and said second logical component[[s]], and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said second logical component[[s]] in response to the generated events indicating the completion of

processing of ~~[[a]]~~ the poll requests by the respective one of said first logical component and said second logical component~~[[s]]~~; and

determining whether the activity period of said first logical component contains the activity period of the second component.

30. (Currently Amended) A method for discovering dependencies between monitored components of a managed data processing system, comprising the steps of:

accessing, from the managed system, run-time activity data for the monitored components, said accessing comprising accessing activity periods for said monitored components, wherein said accessing comprises accessing a first start time and a first end time for a first logical component of said monitored components and a second start time and a second end time for a second logical component of said monitored components;

comparing the accessed run-time activity data of the monitored components to identify positive correlations between the run-time activity of the monitored components, said comparing comprising comparing said activity periods to identify correlations between said monitored components, wherein said comparing comprises determining whether said first start time is before said second start time and whether said first end time is after said second end time; and

generating an identification of a dependency relationship between the first and second logical components for which a positive correlation is identified, wherein ~~the step of monitoring~~ a monitoring agent monitors the run-time activity of said first logical component and said second

logical component ~~comprises generating~~ and generates events in response to completion of the processing of poll requests by each of the first and second logical components, and wherein the step of comparing the monitored run-time activity comprises:

calculating an activity period for each of said first logical component and said second logical component[[s]] in response to generated events indicating the completion of processing of [[a]] the poll requests by the respective one of said first logical component[[s]] and said second logical component[[s]]; and

determining whether the activity period of said first logical component contains the activity period of said second logical component.

31. (Currently Amended) The method according to claim 30, all the limitations of which are incorporated herein by reference, wherein the step of accessing comprises accessing the run-time activity data via an API provided by the managed data processing system.

33. (Currently Amended) The method according to claim 32, all the limitations of which are incorporated herein by reference, further comprising:

computing a value representing the consistency of identification of a positive correlation between the monitored components for the plurality of run-time activity metrics; and storing the computed value in association with the generated identification of a dependency relationship.

34. (Currently Amended) The method according to claim 32, all the limitations of which are incorporated herein by reference, wherein the step of comparing the accessed

run-time activity data comprises comparing run-time activity data for a plurality of executions of the monitored components, and wherein the method further comprises:

computing a single value representing the consistency of identification of a positive correlation between the monitored components, for the plurality of executions of the monitored components and the plurality of run-time activity metrics; and

storing the computed single value in association with the generated identification of a dependency relationship.

35. (Currently Amended) The method according to claim 30, all the limitations of which are incorporated herein by reference, wherein the step of comparing the accessed run-time activity data comprises comparing run-time activity data for a plurality of executions of the monitored components, and wherein the method further comprises:

computing a value representing the consistency of identification of a positive correlation between the monitored components for the plurality of executions of the monitored components; and storing the computed value in association with the generated identification of a dependency relationship.

Examiner's Statement of Reason(s) for Allowance

3. The following is an examiner's statement of reasons for allowance:
4. The closest prior arts of record, i.e. Keller, The Open group, Kar, and Qin, taken alone or in combination, fail to teach or fairly suggest at least: monitoring the run-time activity of the logical components by a monitoring agent that generates events in response to completion of the

processing of a poll request by the logical components, calculating an activity period for the logical components in response to the generated events indicating the completion of processing of the poll request, determining a positive correlation between the logical components, and comparing the start and end times of the logical components, as recited in the independent claims.

While Qin discloses predicting application response time to estimate application performance based on predicting the response time of each thread of the application, Qin ultimately is not related to monitoring the run-time activity of the logical components by a monitoring agent that generates events in response to completion of the processing of a poll request by the logical components. Furthermore, Qin does not disclose calculating an activity period for the logical components in response to the generated events indicating the completion of processing of the poll request, and determining a positive correlation between the logical components.

While Keller and Kar discloses determining dynamic component dependency and the Open group discloses object containment, Keller, Kar, and the Open group do not disclose monitoring the run-time activity of the logical components by a monitoring agent that generates events in response to completion of the processing of a poll request by the logical components, calculating an activity period for the logical components in response to the generated events indicating the completion of processing of the poll request, determining a positive correlation between the logical components, and comparing the start and end times of activity periods of the logical components.

5. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to INSUN KANG whose telephone number is (571)272-3724. The examiner can normally be reached on M-F 8:30-5 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Lewis A. Bullock, Jr. can be reached on 571-272-3759. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Insun Kang/
Examiner, Art Unit 2193

/Lewis A. Bullock, Jr./
Supervisory Patent Examiner, Art Unit 2193